数理解析研究所講究録
229巻 1975年 18-28

コスト関数をもつ有限システム：逐次決定過程

京大・工学部　茨木 俊秀

1．まえがき．　ここでいう逐次決定過程とは，簡単にいえば，有限オートマトンにコスト関数を付与したものである．このモデルは，いわゆる組合せ最適化問題を一般的に表現する手段として研究されてきた．しかし，対象を組合せ最適化問題に限定せず，一つのシステムモデルと考えることも可能であろう．

以下，逐次決定過程に関するこれまでの成果を簡単に紹介するとともに，システムモデルとしての種々の可能性を考察する．

2．諸定義．　$M=(Q, \Sigma, q_0, \lambda, Q_F)$ を有限オートマトン（fa）とする．ただし，

Q：状態の空でない有限集合

Σ：有限アルファベット．$a \in \Sigma$ を決定という．

$q_0 \in Q$：初期状態

$\lambda : Q \times \Sigma \to Q$，状態遷移関数

$Q_F \subset Q$：最終状態の集合．

λは通常の方法で$Q \times \Sigma^* \to Q$へ拡張可能．Σ^*はΣから生成される有限系列（方策という）の集合．

逐次決定過程（sdp, Sequential Decision Process）とはシステム$\Pi = (M, h, \xi_0)$である．ただし，

M：fa

$h : E \times Q \times \Sigma \to E$　（Eは実数の集合）

$\xi_0 \in E$：初期状態q_0のもつ初期コスト値．

hは$h(\xi, q, xa) = h(h(\xi, q, x), \lambda(q, x), a)$によって$E \times Q \times \Sigma^* \to E$に拡張できる．

$h(\xi, q, a)$がξに関し単調ならばΠはmsdp（Monotone sdp）であるという．

また，$F(\Pi) = \{x \in \Sigma^* \mid \lambda(q_0, x) \in Q_F\}$を$\Pi$が受理する方策の集合，$O(\Pi) = \{x \in F(\Pi) \mid (\forall y \in F(\Pi))(h(\xi_0, q_0, x) \leq h(\xi_0, q_0, y))\}$を最適方策の集合という．

3．組合せ最適化問題のsdpによる表現．　簡単のため例を用いて説明する．

行商人問題：n個の節点を一巡する最短閉路を求めよ．た

だし，節点 i から節点 j への距離を c_{ij} とする．

この問題に対し，つぎの sdp $\Pi = (M(Q, \Sigma, q_0, \lambda, Q_F), h, \xi_0)$ を考える（出発点を節点 1 とする）．

$$Q = \{ [S, i] \mid S \subset \{2, 3, \dots, n\},\ i \in S \} \cup \{ [\phi, 1], q_d, q_F \}$$

$$\Sigma = \{ a_1, a_2, \dots, a_n \}$$ （a_i は"つぎに i へ進む"決定）

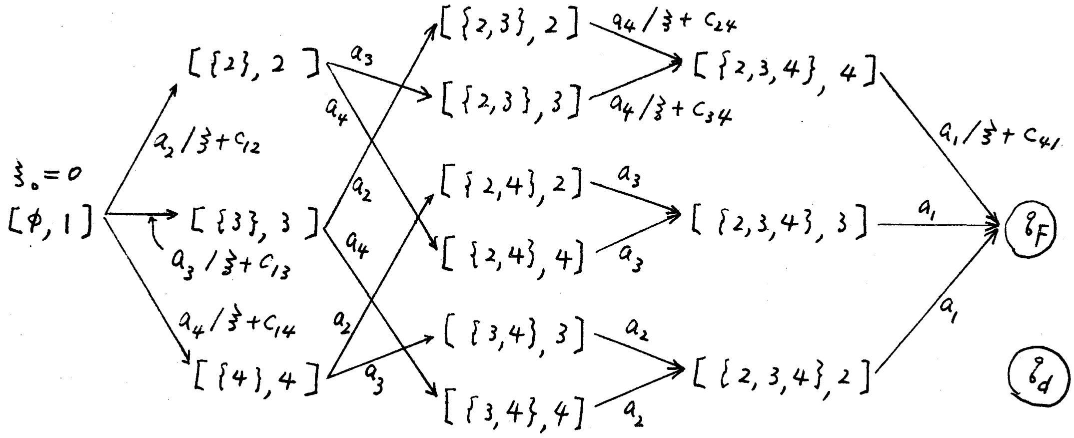
$$q_0 = [\phi, 1], \quad Q_F = \{ q_F \}$$

$$\lambda([S, i], a_j) = \begin{cases} [S \cup \{j\}, j] & \text{if } j \notin S \wedge j \neq 1 \\ q_d & \text{if } j \in S \text{ or } S \neq \{2, \dots, n\} \wedge j = 1 \\ q_F & \text{if } S = \{2, \dots, n\} \wedge j = 1 \end{cases}$$

$$\lambda(q_d, a_j) = \lambda(q_F, a_j) = q_d$$

$$h(\xi, [S, i], a_j) = \xi + c_{ij}, \quad \xi_0 = 0.$$

$n = 4$ の場合の例を下図に示す



（各枝に $a / h(\xi, q, a)$ を付す（h は一部省略）．q_d への状態遷移も省略．）

Πはつぎの意味で行商人問題を表現している．1）$x \in F(\Pi)$ $\Leftrightarrow$ xはn節点を一巡する閉路を表わす．2）$x \in F(\Pi)$に対して$h(\xi_0, \delta_0, x)$は閉路の長さを与える．

sdp Πは，M, h, ξ_0をうまく定めることによって，種々の組合せ最適化問題を表現できる[1]．表現問題に関しては[1][2]に詳しい議論がある．[3]はより一般的な前提で論じている．また，(状態数)最小表現が[4][5]で扱われている．

4．<u>最適方策</u>．　sdpに対しその最適方策を求めるアルゴリズムを与えることができれば，sdpに表現可能なすべての組合せ最適化問題のアルゴリズムとして使用できる．この観点からmsdpは重要である．なぜなら，msdpに表現可能なことと，動的計画法（DP）における最適性の原理が成立することを同一視できるからである[1]．

しかし，任意のmsdpに対して，その最適方策の集合$O(\Pi)$あるいは最適方策$x \in O(\Pi)$を求める一般的なアルゴリズムは存在しない*[6]．アルゴリズムの存在を保障するにはさらに制約を加える必要がある．そのようなクラスとしてつぎのも

* アルゴリズムの存在を論じる際には，$h: Z \times Q \times \Sigma \to Z$（$Z$は整数の集合）とし，$h$を帰納関数と仮定する等の変更を要するが，詳細は略し区別せずに用いる．

のが代表的である.

pmsdp (Positively msdp) : msdp $\wedge$ $h(\xi, q, a) \geq \xi$

smsdp (Strictly msdp) : $\xi_1 > \xi_2 \Rightarrow h(\xi_1, q, a) > h(\xi_2, q, a)$

lmsdp (Loop-Free msdp) : msdp $\wedge$ $|F(\Pi)| < \infty$.

具体的なアルゴリズムは [7] 参照.

また, 各クラスに対する $O(\Pi) \subset \Sigma^*$ の性質を明らかにすることはそれ自身興味ある問題であり, ある程度の結果が得られている [2][6][7]. 閉包性に関する結果を次表にまとめる. (なお, dfs 以下は §6.1 参照)

	$U\cap V$	$U\cup V$	$\overline{U}$	UV	$\min U$		$U\cap V$	$U\cup V$	$\overline{U}$	UV	$\min U$
sdp	o	v	v	v	o	dfs	o	v	v	v	o
msdp	v	v	v	v	v	mdfs	v	v	v	v	v
smsdp	o	o	o	o	o	smdfs	o	v	v	v	o
pmsdp	o	o	o	o	o	pmdfs	v	v	v	v	o
lmsdp	o	o	v	o	o	lmdfs	o	o	v	o	o

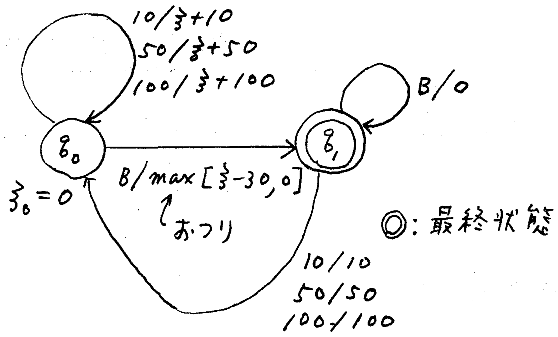
v : 閉じていない, o : 閉じている.

smsdp, pmsdp においては, $U = O(\Pi)$ をみたす Π が存在する必要十分条件は U が正規集合になることである.

5. <u>システムモデルとしての sdp</u>. この場合, 関数 h をシステム動作のコスト, 何らかの Figure of Merit, あるいは記憶内容を表わす一つのパラメータと解釈できよう.

例. 自動入場券販売機:

$\Sigma=\{10$ (10円入れる), 50 (50円), 100 (100円), B (ボタンを押す)$\}$



この立場に立つとき, $O(\Pi)$以外の性質にも注目する必要がある. たとえば, ハードウェアで実現するための理論, 合成・分解, あるコスト値をもつ方策集合の性質, 同定, 可制御・可観測性 etc. である. 成果次第ではsdpをディジタルシステムの有力な表現手段とできよう.

6. 概念の拡張.

6.1 時間依存コスト関数.

一つの可能性は, 時変システムへの拡張である. 一例として, h を $E\times Q\times\Sigma\times T\rightarrow E$ とした離散的有限システム (dfs, Discrete Finite System) がある. ただし, $T=\{0, 1, \dots\}$ は離散時刻の集合である. [9][10]の結果によれば, 表現問題に関してはsdpの結果をそのまま拡張できるが, $O(\Pi)$の性質, たとえばその閉包性については前表のように相当変化する (dfsの部分クラスもsdpと同様に定義する). また, このモデルではほとんどの決定問題が時変性のために決定不能となり, あまり意味のある結果は得られないようである.

6.2 <u>コストに依存した状態遷移</u>. 状態遷移関数を $\lambda : Q \times \Sigma \times E \to Q$, すなわちその時点のコスト値 ξ によって遷移先が変動することを許せば, システムの表現能力が飛躍的に増大する.

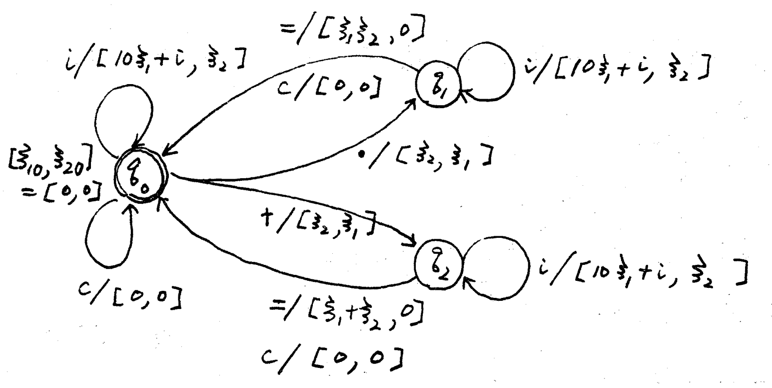
しかし, このモデルは, 適当な仮定の下で, Turing機械と同じ能力をもつことになり, 理論的に目新しい結果は得られないようである. すなわち, M としてTuring機械の制御部, コスト値 ξ としてその時点のテープの内容を数値化したものを想定すればよい.

なお, このモデルはプログラム理論との関連においても興味深い.

6.3 <u>多重コスト関数</u>. コスト関数として, $h_1, h_2, \ldots, h_k$ の k 個を考えることもできる. 一般には, $h_1, h_2, \ldots, h_k$ を一つの h に数値化 (e.g., Gödel Numbering) できるから, sdp としての能力は変化しないことに注意する必要がある. しかし, 表現の簡潔性, コスト関数の単調性などについては本質的な影響がある. また, $O(\pi)$ の定義も一つのポイントになる.

なお, このモデルは, $h_1, h_2, \ldots, h_k$ をレジスタの内容と見ると理解しやすく, ディジタルシステムの便利な表現手段となる可能性がある.

例. 電卓（整数の加算・乗算のみを行なう）：



$i = 0, 1, \dots, 9$

$+$: 加算

$\cdot$: 乗算

c : Clear

$=$: 等号

（たとえば，$123 \cdot 231 = + 13 =$ と押すと $123 \times 231 + 13$ が ξ_1 に入る．）各枝には $a/[h_1(\xi_1, q, a), h_2(\xi_2, q, a)]$ が付されている．

6.4 非決定性状態遷移．非決定性有限オートマトンにならって，$\lambda : Q \times \Sigma \to 2^Q$ とする．すなわち，$\lambda(q, a) = \{q_{i_1}, q_{i_2}, \dots, q_{i_k}\}$ の形になる．コスト関数は遷移先に応じて $h_1(\xi, q, a), \dots, h_k(\xi, q, a)$ とするのが自然であろう．$x \in \Sigma^*$ による遷移 $\lambda(q_0, x)$ はいくつかの経路 $\pi_1, \dots, \pi_s$ で表わされ，そのそれぞれに対しコスト値 $h_{\pi_1}(\xi_0, q_0, x), \dots, h_{\pi_s}(\xi_0, q_0, x)$ が定まる．h および $O(\Pi)$ をつぎのように定義する．

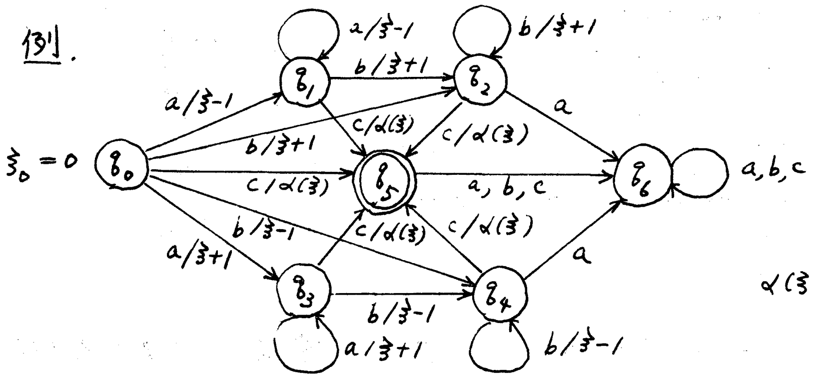
$$h(\xi_0, q_0, x) = \begin{cases} \infty & \text{if } \lambda(q_0, x) \cap Q_F = \phi \\ \min\{h_{\pi_i}(\xi_0, q_0, x) \mid \pi_i \text{は } q \in Q_F \text{ に至る}\}, & \text{その他} \end{cases}$$

$$O(\Pi) = \{x \in \Sigma^* \mid (\forall y \in \Sigma^*)(h(\xi_0, q_0, x) \leq h(\xi_0, q_0, y))\}.$$

以上のモデルを非決定性 sdp と呼ぶ．このとき，任意の非

決定性 sdp Π に対し，$O(\Pi')=O(\Pi)$ をみたす（決定性）sdp が存在する．しかし，この性質を msdp に拡張することはできない．

例．



$$\alpha(\xi)=\begin{cases}1 & \text{if } \xi\ge 0\\ 0 & \text{if } \xi<0\end{cases}$$

明らかに $O(\Pi)=\{a^r b^s c \mid r\neq s\}$ かつ各コスト関数は単調であるが，この $O(\Pi)$ は msdp の最適方策の集合とはならない．
なお，本モデルは阿部のペナルティオートマトン[11]，磯道の Parallel Machine [12] などを特殊な場合として含むという意味でも興味がある．

6.5　確率的状態遷移．　状態遷移を確率的にした確率 sdp を考える．すなわち，各 $a\in\Sigma$ に対し $n\times n$ 確率行列 $P^a=[p_{ij}^a]$ およびコスト行列 $H^a=[h_{ij}^a]$ が定められ，初期確率ベクトル p_0 および最終状態ベクトル $\eta=[\eta_j]$（$\eta_j=1$ if $q_j\in Q_F$，0 if $q_j\notin Q_F$）に対し，$x=a_1a_2\cdots a_t$ が

$$p_0 P^{a_1}P^{a_2}\cdots P^{a_t}\eta \ge c \quad \text{（定数）}$$

を満たせば Π に受理されるとする．x による遷移は n^t 個の経路で表わされ，各経路 π に対し π をとる確率 p_π が計算でき，$h_\pi(\xi_0,q_0,x)$ も非決定性 sdp と同様に定まる．そこで，

$$h(\xi_0,q_0,x)=\sum_\pi p_\pi h_\pi(\xi_0,q_0,x)$$

$$O(\Pi)=\{x\in F(\Pi)\mid(\forall y\in F(\Pi))(h(\xi_0,q_0,x)\le h(\xi_0,q_0,y))\}$$

と定義する.

このモデルはマルコフ決定過程Π'と類似しているが，つぎの点で異なる．1）Π'はQ_Fをもたない，つまり$F(\Pi)=\Sigma^*$. 2）Π'は通常,定常状態での平均コストを問題にする．3）Π'では状態観測可能であり，各状態に対する決定を考える．4）Π'のコスト関数は，通常，$h(\xi, q, a)=\xi+c_{qa}$に限定される.

確率 sdp における $O(\Pi)$ の性質，$O(\Pi)$ を求めるアルゴリズム等に関しては今後の研究に待たねばならない.

6.6 Fuzzy 状態遷移.

状態遷移に Fuzzy 性をもたせると Fuzzy sdp が得られる．定義自体は確率 sdp にならってできるが，その性質についてはほとんど未知である.

文献

[1] R. M. Karp and M. Held, Finite-state processes and dynamic programming, SIAM J. on Applied Math. 15(1967) 693-718.

[2] T. Ibaraki, Representation theorems for equivalent optimization problems, Information and Control 21(1972), 397-435; 茨木，有限状態逐次決定過程の基礎理論と動的計画法への応用，信学論 55-D (1972), 419-426.

[3] -----, Finite state representations of discrete optimization problems, SIAM J. Computing 2(1973), 193-822.

[4] -----, Minimal representations of some classes of dynamic programming, to appear in Information and Control.

[5] 田中，茨木，長谷川，ある種の有限状態最適化問題の最小表現，信学論 57-D (1974), 637-644.

[6] T. Ibaraki, Classes of discrete optimization problems and their decision problems, J. Computer and System Sciences 8(1974), 84-116.

[7] -----, Solvable classes of discrete dynamic programming, J. Mathematical Analysis and Applications 43(1973), 642-693; 茨木，動的計画法の可解なクラス，信学論 55-D (1972), 738-745；茨木，ある種の離散的最適化問題の有限状態表現について，信学論 55-D (1972), 815-822.

[8] -----, On the optimality of algorithms for finite state sequential decision processes, to appear in J. Math. Analysis and Applications.

[9] -----, 離散的有限システムの最適制御，信学論 57-D (1974), 85-92.

[10] -----, 離散的有限システムのいくつかの部分クラス，信学論 58-D (1975), 掲載予定.

[11] 阿部，Penalty automatonによる記号系列パターンの識別，本講究録.

[12] 磯道，時系列パターンのセグメンテーションフリーな認識，本講究録.